Traffic Lights System

by Healthbar Games

LEGAL INFO AND LINKS

Licence:	https://unity3d.com/legal/as_terms
Asset Store link:	https://assetstore.unity.com/packages/slug/124136
Project page:	http://healthbargames.pl/projects/traffic-lights-system
General contact:	contact@healthbargames.pl
Support contact:	support@healthbargames.pl

DESCRIPTION

Traffic Lights System adds simulations of traffic light modules to your game or other 3D applications.

It consists of three components:

- Manager controlling actuation of the successive phases of the simulation
- A component representing a single phase
- A component representing the traffic light module containing red, yellow and green light

The simulation is divided into user defined phases. Each phase groups traffic light modules which should be switched on at the same time and defines timings for lights.

Each phase from the start to the end, goes through the sequence of states:

1. PrepareToGo - lights up red and yellow light
2. Go - lights up green light
3. PrepareToStop - lights up yellow light
4. Stop - lights up red light

Duration of each stage can be set independently for each phase.

Upon completion of the current phase, the manager waits for a defined period of time and starts the next phase from the list of phases. At the end of the list, manager restarts to the first phase and the whole process starts again.

In addition, the manager has two modes (two programs):

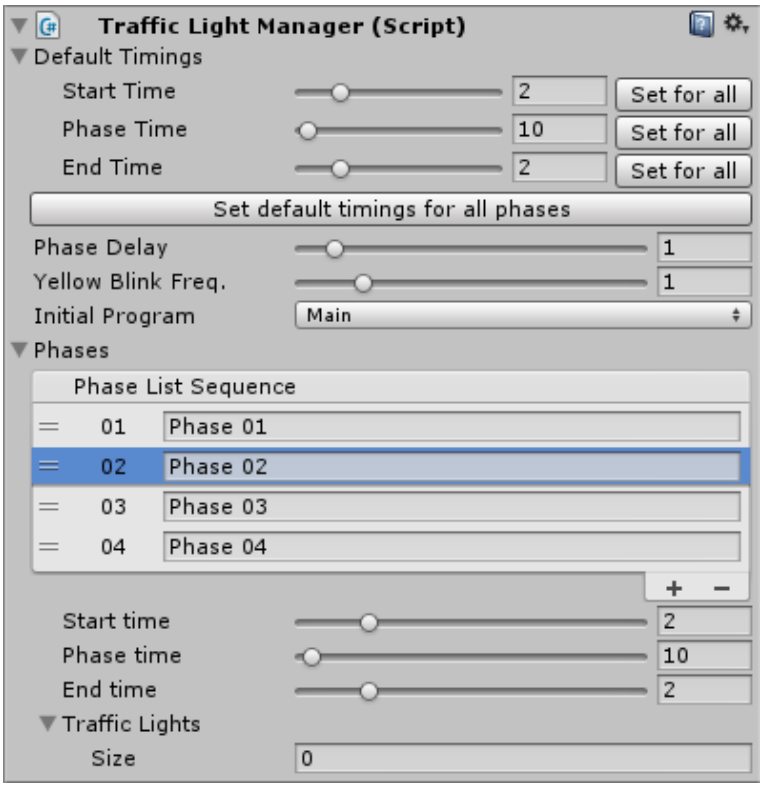
- Main - normal operation
- Malfunction - failure (all light modules from all phases have yellow light blinking)

You can specify the default program which is run by the manager after the start of the scene and change it from a script (as it was used in the demonstration scene).

COMPONENTS

Traffic Light Manager

This is main coordinator for controlling traffic lights modules. One manager can control many different crossroads or you can create different managers for each crossroad in your scene. It is easier to setup when you use one crossroad – one manager model because then you can prepare phase list for lights from one crossroads at a time and don't consider light modules from other crossroads. This also allows to change a program of traffic light system for single crossroad.



Default Timings – section where you can define (and assign for all created phases) default timings for all stages of phase.

Property	Function
Start Time	Duration of start stage (lighting red and yellow light) defined in seconds. [Set for all] button sets defined start time for all phases.

Phase Time	Duration of main stage (lighting green light) defined in seconds. [Set for all] button sets defined phase time for all phases.
End Time	Duration of end stage (lighting yellow light) defined in seconds. [Set for all] button sets defined end time for all phases.
Set default timings for all phases	This button sets all default timings for all phases.
Phase Delay	Delay (in seconds) between phases
Yellow Blink Freq.	Frequency (x times per second) of yellow light blinking during malfunction program.
Initial Program	Program that will be activated after start of the scene.

Phases – section where you can create simulation phases. Phases are numbered and can be rearranged by dragging icons on the left side of the list. Elements of the list can be added using button with plus sign and selected element of list can be deleted using button with minus sign. Each phase can be renamed by clicking on its name field.

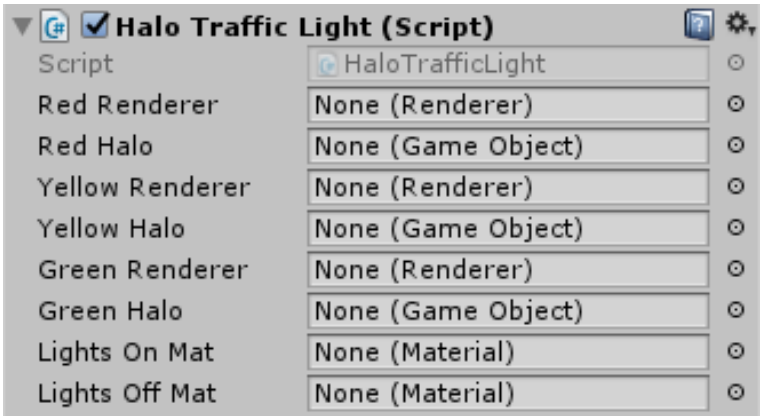
Followed properties are for selected phase and are displayed only when any phase is selected.

Property	Function
Start Time	Duration of start stage (lighting red and yellow light) defined in seconds.
Phase Time	Duration of main stage (lighting green light) defined in seconds.
End Time	Duration of end stage (lighting yellow light) defined in seconds.
Traffic Lights	List of traffic light modules connected to selected phase. All connected light modules are always synchronized with stage of phase they are connected to. When particular phase is in “PrepareToGo” state – all connected light modules will have red and yellow lights turned on, when phase is in “Go” state – all connected light modules will have green light turned on – and so on. You can think of it this way: one phase has to have connected all light modules that have to have green light turned on at the same time. To add traffic light module to the list simply drag game object containing script derived from “TrafficLight” from scene or hierarchy to the list title. This will add another row in the list and attach dragged object. You can also increase size of the list which will add empty row and drag game object to this empty row.

Halo Traffic Light

This component represents single traffic light module with red, yellow and green light. It is derived from abstract class TrafficLightBase which has one abstract function for notifying this component about lights states changes. Components derived from TrafficLightBase need to override this function and when it is called it needs to setup whatever light rig is used according to parameters of this function.

HaloTrafficLight uses light rig that has three renderers for changing light model material when particular light is turning on or off. It also has three game objects with Halo component (to add feeling that lights are glowing when they are turned on).



Property	Function
Red Renderer	Renderer of red light game object
Red Halo	Red light game object with Halo effect component
Yellow Renderer	Renderer of yellow light game object
Yellow Halo	Yellow light game object with Halo effect component
Green Renderer	Renderer of green light game object
Green Halo	Green light game object with Halo effect component
Lights On Mat	Material used when light (red, yellow or green) is turned on
Lights Off Mat	Material used when light (red, yellow or green) is turned off

Real Traffic Light

This component also represents single traffic light module with red, yellow and green light and is very similar to HaloTrafficLight. However, its rig (besides renderers with changing textures and Halo components) has additional game objects with real lights (Point lights in this case). This module has better lighting impressions but at the expense of application performance.

Property	Function
Red Renderer	Renderer of red light game object
Red Halo	Red light game object with Halo effect component
Red Light	Red light game object with Light component (e.g. Point or Spot light)
Yellow Light Renderer	Renderer of yellow light game object
Yellow Light Halo	Yellow light game object with Halo effect component
Yellow Light	Yellow light game object with Light component (e.g. Point or Spot light)
Green Light Renderer	Renderer of green light game object
Green Light Halo	Green light game object with Halo effect component
Green Light	Green light game object with Light component (e.g. Point or Spot light)
Lights On Mat	Material used when light (red, yellow or green) is turned on
Lights Off Mat	Material used when light (red, yellow or green) is turned off